# Multiclass Learning with Simplex Coding

Youssef Mroueh[♯,‡], Tomaso Poggio[♯], Lorenzo Rosasco[♯,‡] Jean-Jacques E. Slotine[†]
♯ - *CBCL, McGovern Institute, MIT*;† - IIT; † - *ME, BCS, MIT*
`ymroueh, lrosasco,jjs@mit.edu tp@ai.mit.edu`

September 17, 2012

**Abstract**

In this paper we discuss a novel framework for multiclass learning, defined by a suitable coding/decoding strategy, namely the simplex coding, that allows to generalize to multiple classes a relaxation approach commonly used in binary classification. In this framework, a relaxation error analysis can be developed avoiding constraints on the considered hypotheses class. Moreover, we show that in this setting it is possible to derive the first provably consistent regularized method with training/tuning complexity which is *independent* to the number of classes. Tools from convex analysis are introduced that can be used beyond the scope of this paper.

## 1 Introduction

As bigger and more complex datasets are available, multiclass learning is becoming increasingly important in machine learning. While theory and algorithms for solving binary classification problems are well established, the problem of multicategory classification is much less understood. Practical multiclass algorithms often reduce the problem to a collection of binary classification problems. Binary classification algorithms are often based on a *relaxation approach*: classification is posed as a non-convex minimization problem and hence relaxed to a convex one, defined by suitable convex loss functions. In this context, results in statistical learning theory quantify the error incurred by relaxation and in particular derive *comparison inequalities* explicitly relating the excess misclassification risk with the excess expected loss, see for example [2, 27, 14, 29] and [18] Chapter 3 for an exhaustive presentation as well as generalizations.

Generalizing the above approach and results to more than two classes is not straightforward. Over the years, several computational solutions have been proposed (among others, see [10, 6, 5, 25, 1, 21]. Indeed, most of the above methods can be interpreted as a kind of relaxation. Most proposed methods have complexity which is more than linear in the number of classes and simple one-vs all in practice offers a good alternative both in terms of performance and speed [15]. Much fewer works have focused on deriving theoretical guarantees. Results in this sense have been pioneered by [28, 20], see also [11, 7, 23]. In these works the error due to relaxation is studied asymptotically and under constraints on the function class to be considered. More quantitative results in terms of comparison inequalities are given in [4] under similar restrictions (see also [19]). Notably, the above results show that seemigly intuitive extensions of binary classification algorithms might lead to methods which are not consistent. Further, it is interesting to note that these restrictions on the function class, needed to prove the theoretical guarantees, make the computations in the corresponding algorithms more involved and are in fact often ignored in practice.

In this paper we dicuss a novel framework for multiclass learning, defined by a suitable coding/decoding strategy, namely the simplex coding, in which a relaxation error analysis can be developed avoiding constraints on the considered hypotheses class. Moreover, we show that in this framework it is possible to derive the first provably consistent regularized method with training/tuning complexity which is *independent* to the number of classes. Interestingly, using the simplex coding, we can naturally generalize results, proof techniques and methods from the binary case, which is recovered as a special case of our theory. Due to space restriction in this paper we focus on extensions of least squares, and SVM loss functions, but our analysis can be generalized to large class of simplex loss functions, including extension of logistic and exponential loss functions (used in boosting). Tools from convex analysis are developed in the longer version of the paper and can be useful beyond the scopes of this paper, and in particular in structured prediction.

The rest of the paper is organized as follow. In Section 2 we discuss problem statement and background. In Section 3 we discuss the simplex coding framework that we analyze in Section 4. Algorithmic aspects and numerical experiments are discussed in Section 5 and Section 6, respectively. Proofs and supplementary technical results are given in the longer version of the paper.

## 2 Problem Statement and Previous Work

Let $(X, Y)$ be two random variables with values in two measurable spaces $\mathcal{X}$ and $\mathcal{Y} = \{1 \dots T\}$, $T \geq 2$. Denote by $\rho_{\mathcal{X}}$, the law of $X$ on $\mathcal{X}$, and by $\rho_j(x)$, the conditional probabilities for $j \in \mathcal{Y}$. The data is a sample $S = (x_i, y_i)_{i=1}^n$, from $n$ identical and independent copies of $(X, Y)$.We can think of $\mathcal{X}$ as a set of possible inputs and of $\mathcal{Y}$, as a set of labels describing a set of semantic categories/classes the input can belong to. A classification rule is a map $b : \mathcal{X} \to \mathcal{Y}$, and its error is measured by the misclassification risk $R(b) = \mathbb{P}(b(X) \neq Y) = \mathbb{E}(\mathbb{1}_{[b(x)\neq y]}(X, Y))$. The optimal classification rule that minimizes $R$ is the Bayes rule, $b_\rho(x) = \arg\max_{y\in\mathcal{Y}} \rho_y(x), x \in \mathcal{X}$. Computing the Bayes rule by directly minimizing the risk $R$, is not possible since the probability distribution is unknown. In fact one could think of minimizing the empirical risk (ERM), $R_S(b) = \frac{1}{n}\sum_{i=1}^n \mathbb{1}_{[b(x)\neq y]}(x_i, y_i)$, which is an unbiased estimator of the $R$, but the corresponding optimization problem is in general not feasible. In binary classification, one of the most common way to obtain computationally efficient methods is based on a relaxation approach. We recall this approach in the next section and describe its extension to multiclass in the rest of the paper.

**Relaxation Approach to Binary Classification.** If $T = 2$, we can set $\mathcal{Y} = \pm 1$. Most modern machine learning algorithms for binary classification consider a convex relaxation of the ERM functional $R_S$. More precisely: 1) the indicator function in $R_S$ is replaced by non negative loss $V : \mathcal{Y} \times \mathbb{R} \to \mathbb{R}^+$ which is convex in the second argument and is sometimes called a *surrogate* loss; 2) the classification rule $b$ replaced by a real valued measurable function $f : \mathcal{X} \to \mathbb{R}$. A classification rule is then obtained by considering the sign of $f$. It often suffices to consider a special class of loss functions, namely large margin loss functions $V : \mathbb{R} \to \mathbb{R}^+$ of the form $V(-yf(x))$. This last expression is suggested by the observation that the misclassification risk, using the labels ±1, can be written as $R(f) = \mathbb{E}(\Theta(-Yf(X)))$, where $\Theta$ is the heavy side step function. The quantity $m = -yf(x)$, sometimes called the *margin*, is a natural point-wise measure of the classification error. Among other examples of large margin loss functions (such as the logistic and exponential loss), we recall the hinge loss $V(m) = |1 + m|_+ = \max\{1 + m, 0\}$ used in support vector machine, and the square loss $V(m) = (1 + m)^2$ used in regularized least squares (note that $(1 - yf(x))^2 = (y - f(x))^2$). Using surrogate large margin loss functions it is possible to design effective learning algorithms replacing the empirical risk with regularized empirical risk minimization

$$\mathcal{E}_S^\lambda(f) = \frac{1}{n}\sum_{i=1}^n V(y_i, f(x_i)) + \lambda\mathcal{R}(f), \tag{1}$$

where $\mathcal{R}$ is a suitable regularization functional and $\lambda$ is the regularization parameter, see Section 5.

### 2.1 Relaxation Error Analysis

As we replace the misclassification loss with a convex *surrogate*— loss, we are effectively changing the problem: the misclassification risk is replaced by the expected loss, $\mathcal{E}(f) = \mathbb{E}(V(-Yf(X)))$ . The expected loss can be seen as a functional on a large space of functions $\mathcal{F} = \mathcal{F}_{V,\rho}$, which depend on $V$ and $\rho$. Its minimizer, denoted by $f_\rho$, replaces the Bayes rule as the target of our algorithm.

The question arises of the price we pay by a considering a relaxation approach: "What is the relationship between $f_\rho$ and $b_\rho$?" More generally, "What is the approximation we incur into by estimating the expected risk rather than the misclassification risk?" The *relaxation error* for a given loss function can be quantified by the following two

requirements:
1) *Fisher Consistency.* A loss function is Fisher consistent if $\text{sign}(f_\rho(x)) = b_\rho(x)$ almost surely (this property is related to the notion of classification-calibration [2]).
2) *Comparison inequalities.* The excess misclassification risk, and the excess expected loss are related by a comparison inequality

$$R(\text{sign}(f)) - R(b_\rho) \leq \psi(\mathcal{E}(f) - \mathcal{E}(f_\rho)),$$

for any function $f \in \mathcal{F}$, where $\psi = \psi_{V,\rho}$ is a suitable function that depends on $V$, and possibly on the data distribution. In particular $\psi$ should be such that $\psi(s) \to 0$ as $s \to 0$, so that if $f_n$ is a (possibly random) sequence of functions, such that $\mathcal{E}(f_n) \to \mathcal{E}(f_\rho)$ (possibly in probability), then the corresponding sequences of classification rules $c_n = \text{sign}(f_n)$ is Bayes consistent, i.e. $R(c_n) \to R(b_\rho)$ (possibly in probability). If $\psi$ is explicitly known, then bounds on the excess expected loss yields bounds on the excess misclassification risk.
The relaxation error in the binary case has been thoroughly studied in [2, 14]. In particular, Theorem 2 in [2] shows that if a large margin surrogate loss is convex, differentiable and decreasing in a neighborhood of $0$, then the loss is Fisher consistent. Moreover, in this case it is possible to give an explicit expression of the function $\psi$. In particular, for the hinge loss the target function is exactly the Bayes rule and $\psi(t) = |t|$. For least squares, $f_\rho(x) = 2\rho_1(x) - 1$, and $\psi(t) = \sqrt{t}$. The comparison inequality for the square loss can be improved for a suitable class of probability distribution satisfying the so called Tsybakov noise condition [22], $\rho_{\mathcal{X}}(\{x \in \mathcal{X}, |f_\rho(x)| \leq s\}) \leq B_q s^q, s \in [0,1], q > 0$. Under this condition the probability of points such that $\rho_y(x) \sim \frac{1}{2}$ decreases polynomially. In this case the comparison inequality for the square loss is given by $\psi(t) = c_q t^{\frac{q+1}{q+2}}$, see [2, 27].
**Previous Works in Multiclass Classification.** From a practical perspective, over the years, several computational solutions to multiclass learning have been proposed. Among others, we mention for example [10, 6, 5, 25, 1, 21]. Indeed, most of the above methods can be interpreted as a kind of relaxation of the original multiclass problem. Interestingly, the study in [15] suggests that the simple one-vs all schemes should be a practical benchmark for multiclass algorithms as it seems to experimentally achive performances that are similar or better to more sophisticated methods.
As we previously mentioned from a theoretical perspective a general account of a large class of multiclass methods has been given in [20], building on results in [2] and [28]. Notably, these results show that seemingly intuitive extensions of binary classification algorithms might lead to *inconsistent* methods. These results, see also [11, 23], are developed in a setting where a classification rule is found by applying a suitable prediction/decoding map to a function $f : \mathcal{X} \to \mathbb{R}^T$ where $f$ is found considering a loss function $V : \mathcal{Y} \times \mathbb{R}^T \to \mathbb{R}^+$. The considered functions have to satisfy the constraint $\sum_{y \in \mathcal{Y}} f^y(x) = 0$, for all $x \in \mathcal{X}$. The latter requirement is problematic since it makes the computations in the corresponding algorithms more involved and is in fact often ignored, so that practical algorithms often come with no consistency guarantees. In all the above papers relaxation is studied in terms of Fisher and Bayes consistency and the explicit form of the function $\psi$ is not given. More quantitative results in terms of explicit comparison inequality are given in [4] and (see also [19]), but also need to to impose the "sum to zero" constraint on the considered function class.

## 3 A Relaxation Approach to Multicategory Classification

In this section we propose a natural extension of the relaxation approach that avoids constraining the class of functions to be considered, and allows to derive explicit comparison inequalities. See Remark 1 for related approaches.

**Simplex Coding.** We start considering a suitable coding/decoding strategy. A *coding* map turns a label $y \in \mathcal{Y}$ into a code vector. The corresponding *decoding* map given a vector returns a label in $\mathcal{Y}$. Note that, this is what we implicitly did while treating binary classification *encoding* the label space $\mathcal{Y} = \{1, 2\}$ using the coding $\pm 1$, so that the naturally decoding strategy is simply $\text{sign}(f(x))$. The coding/decoding strategy we study is described by the following definition.

**Definition 1 (Simplex Coding).** *The simplex coding is a map* $C : \mathcal{Y} \to \mathbb{R}^{T-1}$, $\quad C(y) = c_y$*, where the code vectors* $\mathcal{C} = \{c_y \mid y \in \mathcal{Y}\} \subset \mathbb{R}^{T-1}$ *satisfy: 1)* $\|c_y\|^2 = 1$, $\forall y \in \mathcal{Y}$*, 2)*$\langle c_y, c_{y'} \rangle = -\frac{1}{T-1}$*, for* $y \neq y'$ *with* $y, y' \in \mathcal{Y}$*, and*

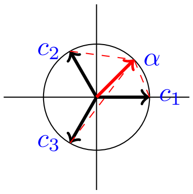


Figure 1: Decoding with simplex coding $T = 3$.

*3)* $\sum_{y\in\mathcal{Y}} c_y = 0$. *The corresponding decoding is the map* $D : \mathbb{R}^{T-1} \to \{1, \dots, T\}$, $\quad D(\alpha) = \arg\max_{y\in\mathcal{Y}} \langle \alpha, c_y \rangle$, $\forall \alpha \in \mathbb{R}^{T-1}$.

The simplex coding corresponds to the $T$ most separated vectors on the hypersphere $\mathbb{S}^{T-2}$ in $\mathbb{R}^{T-1}$, that is the vertices of the simplex (see Figure 1). For binary classification it reduces to the $\pm 1$ coding and the decoding map is equivalent to taking the sign of $f$. The decoding map has a natural geometric interpretation: an input point is mapped to a vector $f(x)$ by a function $f : \mathcal{X} \to \mathbb{R}^{T-1}$, and hence assigned to the class having closer code vector (for $y, y' \in \mathcal{Y}$ and $\alpha \in \mathbb{R}^{T-1}$, we have $\|c_y - \alpha\|^2 \geq \|c_{y'} - \alpha\|^2 \Leftrightarrow \langle c_{y'}, \alpha \rangle \leq \langle c_y, \alpha \rangle$.

**Relaxation for Multiclass Learning.** We use the simplex coding to propose an extension of the binary classification approach. Following the binary case, the relaxation can be described in two steps:

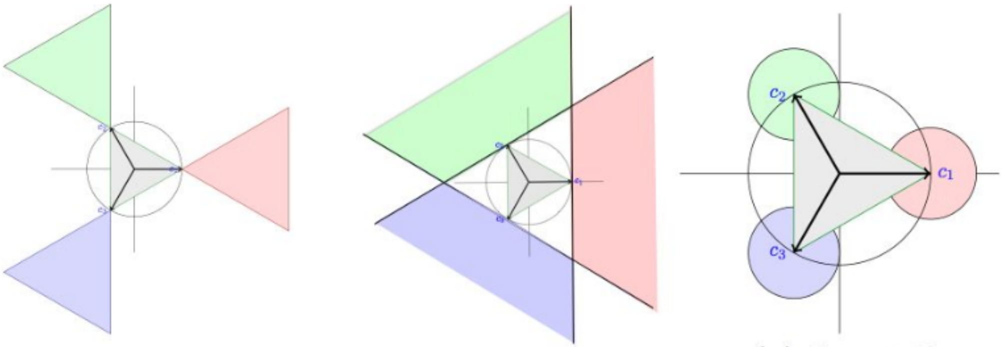
1. using the simplex coding, the indicator function is upper bounded by a non-negative loss function $V : \mathcal{Y} \times \mathbb{R}^{T-1} \to \mathbb{R}^+$, such that $\mathbb{1}_{[b(x)\neq y]}(x, y) \leq V(y, C(b(x)))$,for all $b : \mathcal{X} \to \mathcal{Y}$, and $x \in \mathcal{X}, y \in \mathcal{Y}$,

2. rather than $C \circ b$ we consider functions with values in $f : \mathcal{X} \to \mathbb{R}^{T-1}$, so that $V(y, C(b(x))) \leq V(y, f(x))$, for all $b : \mathcal{X} \to \mathcal{Y}$, $f : \mathcal{X} \to \mathbb{R}^{T-1}$ and $x \in \mathcal{X}, y \in \mathcal{Y}$.

In the next section we discuss several loss functions satisfying the above definitions and we study in particular the extension of the least squares and SVM loss functions.

**Multiclass Simplex Loss Functions.** Several loss functions for binary classification can be naturally extended to multiple classes using the simplex coding. Due to space restriction, in this paper we focus on extensions of least squares, and SVM loss functions, but our analysis can be generalized to large class of simplex loss functions, including extension of logistic and exponential loss functions( used in boosting). The Simplex Least Square loss (**S-LS**) is given by $V(y, f(x)) = \|c_y - f(x)\|^2$, and reduces to the usual least square approach to binary classification for $T = 2$. One natural extension of the SVM's hinge loss in this setting would be to consider the Simplex Half space SVM loss (**SH-SVM**) $V(y, f(x)) = |1 - \langle c_y, f(x) \rangle|_+$. We will see in the following that while this loss function would induce efficient algorithms in general is not Fisher consistent unless further constraints are assumed. In turn, this latter constraint would considerably slow down the computations. Then we consider a second loss function Simplex Cone SVM (**SC-SVM**), related to the hinge loss, which is defined as $V(y, f(x)) = \sum_{y'\neq y} \left| \frac{1}{T-1} + \langle c_{y'}, f(x) \rangle \right|_+$. The latter loss function is related to the one considered in the multiclass SVM proposed in [10]. We will see that it is possible to quantify the relaxation error of the loss function without requiring further constraints. Both the above SVM loss functions reduce to the binary SVM hinge loss if $T = 2$.

**Remark 1** (Related approaches)**.** *The simplex coding has been considered in [8],[26], and [16]. In particular, a kind of SVM loss is considered in [8] where* $V(y, f(x)) = \sum_{y'\neq y} |\varepsilon - \langle f(x), v_{y'}(y) \rangle|_+$ *and* $v_{y'}(y) = \frac{c_y - c_{y'}}{\|c_y - c_{y'}\|}$, *with* $\varepsilon = \langle c_y, v_{y'}(y) \rangle = \frac{1}{\sqrt{2}}\sqrt{\frac{T}{T-1}}$. *More recently [26] considered the loss function* $V(y, f(x)) = |\varepsilon - \|c_y - f(x)\||_+$, *and a simplex multi-class boosting loss was introduced in [16], in our notation* $V(y, f(x)) = \sum_{j\neq y} e^{-\langle c_y - c_{y'}, f(x) \rangle}$. *While all those losses introduce a certain notion of margin that makes use of the geometry of the simplex coding, it is not to clear how to derive explicit comparison theorems and moreover the computational complexity of the resulting algorithms scales linearly with the number of classes in the case of the losses considered in [16, 26] and* $O((nT)^\gamma), \gamma \in \{2, 3\}$ *for losses considered in [8] .*



$(a) SC - SVM$ $(b) SH - SVM$ $(c) S - LS$

Figure 2: Level sets of different losses considered for $T = 3$. A classification is correct if an input $(x, y)$ is mapped to a point $f(x)$ that lies in the neighborhood of the vertex $c_y$. The shape of the neighborhood is defined by the loss, it takes form of a cone supported on a vertex in the case of SC-SVM, a half space delimited by the hyperplane orthogonal to the vertex in the case of the SH-SVM, and a sphere centered on the vertex in the case of S-LS.

## 4 Relaxation Error Analysis

If we consider the simplex coding, a function $f$ taking values in $\mathbb{R}^{T-1}$, and the decoding operator $D$, the misclassification risk can also be written as: $R(D(f)) = \int_{\mathcal{X}} (1 - \rho_{D(f(x))}) d\rho_{\mathcal{X}}(x)$. Then, following a relaxation approach we replace the misclassification loss by the expected risk induced by one of the loss functions $V$ defined in the previous section. As in the binary case we consider the expected loss $\mathcal{E}(f) = \int V(y, f(x)) d\rho(x, y)$. Let $L^p(\mathcal{X}, \rho_{\mathcal{X}}) = \{f : \mathcal{X} \to \mathbb{R}^{T-1} \mid \|f\|_\rho^p = \int \|f(x)\|^p \, d\rho_{\mathcal{X}}(x) < \infty\}$, $p \geq 1$.

The following theorem studies the relaxation error for SH-SVM, SC-SVM, and S-LS loss functions.

**Theorem 1.** *For SH-SVM, SC-SVM, and S-LS loss functions, there exists a $p$ such that $\mathcal{E} : L^p(\mathcal{X}, \rho_{\mathcal{X}}) \to \mathbb{R}^+$ is convex and continuous. Moreover,*

1. *The minimizer $f_\rho$ of $\mathcal{E}$ over $\mathcal{F} = \{f \in L^p(\mathcal{X}, \rho_{\mathcal{X}}) \mid f(x) \in K \ a.s.\}$ exists and $D(f_\rho) = b_\rho$.*
2. *For any $f \in \mathcal{F}$, $R(D(f)) - R(D(f_\rho)) \leq C_T(\mathcal{E}(f) - \mathcal{E}(f_\rho))^\alpha$, where the expressions of $p, K, f_\rho, C_T$, and $\alpha$ are given in Table 1.*

| Loss | $p$ | $K$ | $f_\rho$ | $C_T$ | $\alpha$ |
|---|---|---|---|---|---|
| SH-SVM | 1 | $conv(\mathcal{C})$ | $c_{b_\rho}$ | $T - 1$ | 1 |
| SC-SVM | 1 | $\mathbb{R}^{T-1}$ | $c_{b_\rho}$ | $T - 1$ | 1 |
| S-LS | 2 | $\mathbb{R}^{T-1}$ | $\sum_{y \in \mathcal{Y}} \rho_y c_y$ | $\sqrt{\frac{2(T-1)}{T}}$ | $\frac{1}{2}$ |

Table 1: $conv(\mathcal{C})$ is the convex hull of the set $\mathcal{C}$ defined in (1).

The proof of this theorem is given in the longer version of the paper.

The above theorem can be improved for Least Squares under certain classes of distribution . Toward this end we introduce the following notion of misclassification noise that generalizes Tsybakov's noise condition.

**Definition 2.** *Fix $q > 0$, we say that the distribution $\rho$ satisfy the multiclass noise condition with parameter $B_q$, if*

$$\rho_{\mathcal{X}}\left(\left\{x \in \mathcal{X} \mid 0 \leq \min_{j \neq D(f_\rho(x))} \frac{T-1}{T}(\langle c_{D(f_\rho(x))} - c_j, f_\rho(x)\rangle) \leq s\right\}\right) \leq B_q s^q, \tag{2}$$

*where $s \in [0, 1]$.*

If a distribution $\rho$ is characterized by a very large $q$, then, for each $x \in \mathcal{X}$, $f_\rho(x)$ is arbitrarily close to one of the coding vectors. For $T = 2$, the above condition reduces to the binary Tsybakov noise. Indeed, let $c_1 = 1$, and $c_2 = -1$, if $f_\rho(x) > 0$, $\frac{1}{2}(c_1 - c_2)f_\rho(x) = f_\rho(x)$, and if $f_\rho(x) < 0$, $\frac{1}{2}(c_2 - c_1)f_\rho(x) = -f_\rho(x)$.

The following result improves the exponent of simplex-least square to $\frac{q+1}{q+2} > \frac{1}{2}$ :

**Theorem 2.** *For each $f \in L^2(\mathcal{X}, \rho_{\mathcal{X}})$, if (2) holds, then for S-LS we have the following inequality,*

$$R(D(f)) - R(D(f_\rho)) \leq K\left(\frac{2(T-1)}{T}(\mathcal{E}(f) - \mathcal{E}(f_\rho))\right)^{\frac{q+1}{q+2}}, \tag{3}$$

*for a constant $K = \left(2\sqrt{B_q + 1}\right)^{\frac{2q+2}{q+2}}$.*

**Remark 2.** *Note that the comparison inequalities show a tradeoff between the exponent $\alpha$ and the constant $C(T)$, for S-LS and SVM losses. While the constant is order $T$ for SVM it is order $1$ for S-LS, on the other hand the exponent is $1$ for SVM losses and $\frac{1}{2}$ for S-LS. The latter could be enhanced to $1$ for close to separable classification problems by virtue of the Tsybakov noise condition.*

**Remark 3.** *Comparison inequalities given in Theorems 1 and 2 can be used to derive generalization bounds on the excess misclassification risk. For least square min-max sharp bound, for vector valued regression are easy to derive.Standard techniques for deriving sample complexity bound in binary classification extended for multi-class SVM losses could be found in [7] and could be adapted to our setting. The obtained bound are not known to be tight, better bounds akin to those in [18], will be subject to future work.*

# 5 Computational Aspects and Regularization Algorithms

In this section we discuss some computational implications of the framework we presented.

**Regularized Kernel Methods.** We consider regularized methods of the form (1), induced by simplex loss functions and where the hypotheses space is a vector valued reproducing kernel Hilbert spaces (VV-RKHSs) and the regularizer the corresponding norm. See Appendix D.2 for a brief introduction to VV-RKHSs.

In the following, we consider a class of kernels such that the corresponding RKHS $\mathcal{H}$ is given by the completion of the span $\{f(x) = \sum_{i=j}^{N} \Gamma(x_j, x)a_j,\ a_j \in \mathbb{R}^{T-1}, x_i, \in \mathcal{X},\ \forall j = 1, \ldots, N\}$, where we note that the coefficients are vectors in $\mathbb{R}^{T-1}$. While other choices are possible this is the kernel more directly related to a one vs all approach. We will discuss in particular the case where the kernel is induced by a finite dimensional feature map, $k(x, x') = \langle \Phi(x), \Phi(x')\rangle$, where $\Phi : \mathcal{X} \to \mathbb{R}^p$, and $\langle \cdot, \cdot \rangle$ is the inner product in $\mathbb{R}^p$. In this case we can write each function in $\mathcal{H}$ as $f(x) = W\Phi(x)$, where $W \in \mathbb{R}^{(T-1)\times p}$.

It is known [12, 3] that the representer theorem [9] can be easily extended to a vector valued setting, so that that minimizer of a simplex version of Tikhonov regularization is given by $f_S^\lambda(x) = \sum_{j=1}^{n} k(x, x_j)a_j$, $a_j \in \mathbb{R}^{T-1}$, for all $x \in \mathcal{X}$, where the explicit expression of the coefficients depends on the considered loss function. We use the following notations: $K \in \mathbb{R}^{n\times n}, K_{ij} = k(x_i, x_j), \forall i, j \in \{1 \ldots n\}, A \in \mathbb{R}^{n\times(T-1)}, A = (a_1, ..., a_n)^T$.

**Simplex Regularized Least squares (S-RLS).** S-RLS is obtained considering the simplex least square loss in the Tikhonov functionals. It is easy to see [15] that in this case the coefficients must satisfy either $(K + \lambda nI)A = \hat{Y}$ or $(\hat{X}^T\hat{X} + \lambda nI)W = \hat{X}^T\hat{Y}$ in the linear case, where $\hat{X} \in \mathbb{R}^{n\times p}, \hat{X} = (\Phi(x_1), ..., \Phi(x_n))^\top$ and $\hat{Y} \in \mathbb{R}^{n\times(T-1)}, \hat{Y} = (c_{y_1}, ..., c_{y_n})^\top$ .

Interestingly, the classical results from [24] can be extended to show that the value $f_{S_i}(x_i)$, obtained computing the solution $f_{S_i}$ removing the $i-th$ point from the training set (the leave one out solution), can be computed in closed form. Let $f^{\lambda}_{loo} \in \mathbb{R}^{n\times(T-1)}, f^{\lambda}_{loo} = (f^{\lambda}_{S_1}(x_1), \dots, f^{\lambda}_{S_n}(x_n))$. Let $\mathcal{K}(\lambda) = (K + \lambda n I)^{-1}$and $C(\lambda) = \mathcal{K}(\lambda)\hat{Y}$. Define $M(\lambda) \in \mathbb{R}^{n\times(T-1)}$, such that: $M(\lambda)_{ij} = 1/\mathcal{K}(\lambda)_{ii}$, $\forall\ j = 1 \dots T-1$. One can show similarly to [15], that $f^{\lambda}_{loo} = \hat{Y} - C(\lambda) \odot M(\lambda)$, where $\odot$ is the Hadamard product. Then, the leave-one-out error $\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{y \neq D(f_{S^i}(x))}(y_i, x_i)$, can be minimized at essentially no extra cost by precomputing the eigen decomposition of $K$ (or $\hat{X}^T\hat{X}$).

**Simplex Cone Support Vector Machine (SC-SVM).** Using standard reasoning it is easy to show that (see Appendix C.2), for the SC-SVM the coefficients in the representer theorem are given by $a_i = -\sum_{y\neq y_i} \alpha_i^y c_y, \quad i = 1, \dots, n$, where $\alpha_i = (\alpha_i^y)_{y\in\mathcal{Y}} \in \mathbb{R}^T, i = 1, \dots, n$, solve the quadratic programming (QP) problem

$$\max_{\alpha_1,\dots,\alpha_n \in \mathbb{R}^T} \left\{ -\frac{1}{2} \sum_{y,y',i,j} \alpha_i^y K_{ij} G_{yy'} \alpha_j^{y'} + \frac{1}{T-1} \sum_{i=1}^{n} \sum_{y=1}^{T} \alpha_i^y \right\} \tag{4}$$
$$\text{subject to} \quad 0 \leq \alpha_i^y \leq C_0 \delta_{y,y_i},\ \forall\ i = 1, \dots, n, y \in \mathcal{Y}$$

where $G_{y,y'} = \langle c_y, c_{y'} \rangle\ \forall y, y' \in \mathcal{Y}$ and $C_0 = \frac{1}{2n\lambda}$, $\alpha_i = (\alpha_i^y)_{y\in\mathcal{Y}} \in \mathbb{R}^T$, for $i = 1, \dots, n$ and $\delta_{i,j}$ is the Kronecker delta.

**Simplex Halfspaces Support Vector Machine (SH-SVM).** A similar, yet more more complicated procedure, can be derived for the SH-SVM. Here, we omit this derivation and observe instead that if we neglect the convex hull constraint from Theorem 1, requiring $f(x) \in \text{co}(\mathcal{C})$ for almost all $x \in \mathcal{X}$, then the SH-SVM has an especially simple formulation at the price of loosing consistency guarantees. In fact, in this case the coefficients are given by $a_i = \alpha_i c_{y_i}, \quad i = 1, \dots, n$, where $\alpha_i \in \mathbb{R}$, with $i = 1, \dots, n$ solve the quadratic programming (QP) problem

$$\max_{\alpha_1,\dots,\alpha_n \in \mathbb{R}} -\frac{1}{2} \sum_{i,j} \alpha_i K_{ij} G_{y_i y_j} \alpha_j + \sum_{i=1}^{n} \alpha_i$$
$$\text{subject to} \quad 0 \leq \alpha_i \leq C_0,\ \forall\ i = 1 \dots n,$$

where $C_0 = \frac{1}{2n\lambda}$. The latter formulation could be trained at the same complexity of the binary SVM (worst case $O(n^3)$) but lacks consistency.

**Online/Incremental Optimization** The regularized estimators induced by the simplex loss functions can be computed by mean of online/incremental first order (sub) gradient methods. Indeed, when considering finite dimensional feature maps, these strategies offer computationally feasible solutions to train estimators for large datasets where neither a $p$ by $p$ or an $n$ by $n$ matrix fit in memory. Following [17] we can alternate a step of stochastic descent on a data point : $W_{\text{tmp}} = (1 - \eta_i\lambda)W_i - \eta_i\partial(V(y_i, f_{W_i}(x_i)))$ and a projection on the Frobenius ball $W_i = \min(1, \frac{1}{\sqrt{\lambda}||W_{tmp}||_F})W_{\text{tmp}}$ (See Algorithn C.5 for details.) The algorithm depends on the used loss function through the computation of the (point-wise) subgradient $\partial(V)$. The latter can be easily computed for all the loss functions previously discussed. For the SLS loss we have $\partial(V(y_i, f_W(x_i))) = 2(c_{y_i} - Wx_i)x_i^\top$, while for the SC-SVM loss we have $\partial(V(y_i, f_W(x_i))) = (\sum_{k\in I_i} c_k)x_i^\top$ where $I_i = \{y \neq y_i | \langle c_y, Wx_i \rangle > -\frac{1}{T-1}\}$. For the SH-SVM loss we have: $\partial(V(y, f_W(x_i))) = -c_{y_i}x_i^\top$ if $c_{y_i}Wx_i < 1$ and 0 else .

## 5.1 Comparison of Computational Complexity

The cost of solving S-RLS for fixed $\lambda$ is in the worst case $O(n^3)$ (for example via Choleski decomposition). If we are interested into computing the regularization path for $N$ regularization parameter values, then as noted in [15] it might be convenient to perform an eigendecomposition of the kernel matrix rather than solving the systems $N$ times. For explicit feature maps the cost is $O(np^2)$, so that the cost of computing the regularization path for simplex RLS algorithm is $O(min(n^3, np^2))$ and hence *independent* of $T$. One can contrast this complexity with the one of a näive One Versus all (OVa) approach that would lead to a $O(Nn^3T)$ complexity. Simplex SVMs can be solved using solvers available for binary SVMs that are considered to have complexity $O(n^\gamma)$ with $\gamma \in \{2, 3\}$(actually the complexity scales with the number of support vectors) . For SC-SVM, though, we have $nT$ rather than $n$ unknowns and the complexity is $(O(nT)^\gamma)$. SH-SVM where we omit the constraint, could be trained at the same complexity of the binary SVM (worst case $O(n^3)$) but lacks consistency. Note that unlike for S-RLS, there is no straightforward way to compute the regularization path and the leave one out error for any of the above SVMs . The online algorithms induced by the different simplex loss functions are essentially the same, in particular each iteration depends linearly on the number of classes.

# 6 Numerical Results

We conduct several experiments to evaluate the performance of our batch and online algorithms, on 5 UCI datasets as listed in Table 6, as well as on Caltech101 and Pubfig83. We compare the performance of our algorithms to on versus all svm (libsvm) , as well as the simplex based boosting [16]. For UCI datasets we use the raw features, on Caltech101 we use hierarchical features[1] , and on Pubfig83 we use the feature maps from [13]. In all cases the parameter selection is based either on a hold out (ho) $(80\%$ training $-\ 20\%$ validation$)$ or a leave one out error (loo). For the model Selection of $\lambda$ in S-LS, $100$ values are chosen in the range $[\lambda_{min}, \lambda_{max}]$,(where $\lambda_{min}$ and $\lambda_{\max}$, correspond to the smallest and biggest eigenvalues of $K$). In the case of a Gaussian kernel (rbf) we use a heuristic that sets the width of the gaussian $\sigma$ to the 25-th percentile of pairwise distances between distinct points in the training set. In Table 6 we collect the resulting classification accuracies:

| | Landsat | Optdigit | Pendigit | Letter | Isolet | Ctech | Pubfig83 |
|---|---|---|---|---|---|---|---|
| SC-SVM Online (ho) | 65.15% | 89.57% | 81.62% | 52.82% | 88.58% | 63.33% | 84.70% |
| SH-SVM Online (ho) | 75.43% | 85.58% | 72.54% | 38.40% | 77.65% | 45% | 49.76% |
| S-LS Online (ho) | 63.62% | 91.68% | 81.39% | 54.29% | 92.62% | 58.39% | 83.61% |
| S-LS Batch (loo) | 65.88% | 91.90% | 80.69% | 54.96% | 92.55% | 66.35% | 86.63% |
| S-LS rbf Batch (loo) | **90.15%** | **97.09%** | **98.17%** | **96.48%** | **97.05%** | **69.38%** | **86.75%** |
| SVM batch ova (ho) | 72.81% | 92.13% | 86.93% | 62.78% | 90.59% | 70.13% | 85.97% |
| SVM rbf batch ova (ho) | 95.33% | 98.07% | 98.88% | 97.12% | 96.99% | 51.77% | 85.60% |
| Simplex boosting [16] | 86.65% | 92.82% | 92.94% | 59.65% | 91.02% | − | − |

Table 2: Accuracies of our algorithms on several datasets.

As suggested by the theory, the consistent methods SC-SVM and S-LS have a big advantage over SH-SVM (where we omitted the convex hull constraint) . Batch methods are overall superior to online methods, with online SC-SVM achieving the best results. More generally, we see that rbf S- LS has the best performance among the simplex methods including the simplex boosting [16]. When compared to One Versus All SVM-rbf, we see that S-LS rbf achieves essentially the same performance.

# References

[1] Erin L. Allwein, Robert E. Schapire, and Yoram Singer. Reducing multiclass to binary: a unifying approach for margin classifiers. *Journal of Machine Learning Research*, 1:113–141, 2000.

[2] Peter L. Bartlett, Michael I. Jordan, and Jon D. McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006.

[3] A. Caponnetto and E. De Vito. Optimal rates for regularized least-squares algorithm. *Foundations of Computational Mathematics*, 2006.

[4] D. Chen and T. Sun. Consistency of multiclass empirical risk minimization methods based in convex loss. *Journal of machine learning*, X, 2006.

[5] Crammer.K and Singer.Y. On the algorithmic implementation of multiclass kernel-based vector machines. *JMLR*, 2001.

[6] Thomas G. Dietterich and Ghulum Bakiri. Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, 2:263–286, 1995.

[7] Yann Guermeur. Vc theory of large margin multi-category classiers. *Journal of Machine Learning Research*, 8:2551–2594, 2007.

[1] The data set will be made available upon acceptance.

[8] Simon I. Hill and Arnaud Doucet. A framework for kernel-based multi-category classification. *J. Artif. Int. Res.*, 30(1):525–564, December 2007.

[9] G. Kimeldorf and G. Wahba. A correspondence between bayesian estimation of stochastic processes and smoothing by splines. *Ann. Math. Stat.*, 41:495–502, 1970.

[10] Lee.Y, L.Yin, and Wahba.G. Multicategory support vector machines: Theory and application to the classification of microarray data and satellite radiance data. *Journal of the American Statistical Association*, 2004.

[11] Liu.Y. Fisher consistency of multicategory support vector machines. *Eleventh International Conference on Artificial Intelligence and Statistics, 289-296*, 2007.

[12] C.A. Micchelli and M. Pontil. On learning vector–valued functions. *Neural Computation*, 17:177–204, 2005.

[13] N. Pinto, Z. Stone, T. Zickler, and D.D. Cox. Scaling-up biologically-inspired computer vision: A case-study on facebook. 2011.

[14] M.D. Reid and R.C. Williamson. Composite binary losses. *JMLR*, 11, September 2010.

[15] Rifkin.R and Klautau.A. In defense of one versus all classification. *journal of machine learning*, 2004.

[16] Saberian.M and Vasconcelos .N. Multiclass boosting: Theory and algorithms. In *NIPS 2011*, 2011.

[17] Shai Shalev-Shwartz, Yoram Singer, and Nathan Srebro. Pegasos: Primal estimated sub-gradient solver for svm. In *Proceedings of the 24th ICML*, ICML '07, pages 807–814, New York, NY, USA, 2007. ACM.

[18] I. Steinwart and A. Christmann. *Support vector machines*. Information Science and Statistics. Springer, New York, 2008.

[19] Van de Geer.S Tarigan.B. A moment bound for multicategory support vector machines. *JMLR 9, 2171-2185*, 2008.

[20] A. Tewari and P. L. Bartlett. On the consistency of multiclass classification methods. In *Proceedings of the 18th Annual Conference on Learning Theory*, volume 3559, pages 143–157. Springer, 2005.

[21] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *JMLR*, 6(2):1453–1484, 2005.

[22] Alexandre B. Tsybakov. Optimal aggregation of classifiers in statistical learning. *Annals of Statistics*, 32:135–166, 2004.

[23] Elodie Vernet, Robert C. Williamson, and Mark D. Reid. Composite multiclass losses. In *Proceedings of Neural Information Processing Systems (NIPS 2011)*, 2011.

[24] G. Wahba. *Spline models for observational data*, volume 59 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. SIAM, Philadelphia, PA, 1990.

[25] Weston and Watkins. Support vector machine for multi class pattern recognition. *Proceedings of the seventh european symposium on artificial neural networks*, 1999.

[26] Tong Tong Wu and Kenneth Lange. Multicategory vertex discriminant analysis for high-dimensional data. *Ann. Appl. Stat.*, 4(4):1698–1721, 2010.

[27] Y. Yao, L. Rosasco, and A. Caponnetto. On early stopping in gradient descent learning. *Constructive Approximation*, 26(2):289–315, 2007.

[28] T. Zhang. Statistical analysis of some multi-category large margin classification methods. *Journal of Machine Learning Research*, 5:1225–1251, 2004.

[29] Tong Zhang. Statistical behavior and consistency of classification methods based on convex risk minimization. *The Annals of Statistics*, Vol. 32, No. 1, 56134, 2004.